WHAT IS CLAIMED IS:

1. A method for calculating a potential optimum yield for an entity for a set of demands including both transient demands and group demands comprising:

setting a yield formula for calculating a yield that includes both transient yield from transient individual space and group yield from group individual space and group function space;

determining constraints related to the yield formula;

determining bounds related to the yield formula;

identifying which demands should be accepted in order to optimize the yield, subject to the constraints and the bounds; and

determining the potential optimum yield utilizing the demands identified in the identifying step.

2. The method of Claim 1, wherein the identifying step is performed using mixed integer linear programming techniques.

3. The method of Claim 1, wherein the yield comprises revenue.

4. The method of Claim 1, wherein the yield comprises profit.

5. The method of Claim 1, further comprising assigning a small value as a cost of a transient upgrade, and including the small value in the transient yield.

6. The method of Claim 1, further comprising comparing actual total yield to the potential optimum yield.

7. The method of Claim 1, further comprising:

determining at least one upper bound transient constraint, the upper bound transient constraint ensuring that more transient individual space than available is not assigned.

8. The method of Claim 7, wherein the at least one upper bound transient constraint is $x.AD(i).LS(i).RC(i).RT(i) \leq N(i)$.

9. The method of Claim 5, further comprising:

determining a transient upgrade constraint that ensures that an assigned transient individual space level is at least as high as a requested transient individual space level.

10. The method of Claim 9, wherein the transient upgrade constraint is:

$$\sum_{ART \geq RT(i)} y.AD(i).LS(i).RC(i).RT(i).ART \geq x.AD(i).LS(i).RC(i).RT(i).$$

11. The method of Claim 1, further comprising:

determining a transient yield constraint, incorporating transient individual space yield information.

12. The method of Claim 11, wherein the transient yield constraint is

$$\sum_{i} Y(i)x.AD(i).LS(i).RC(i).RT(i)$$

wherein Y(i) is the yield associated with transient demand i.

13. The method of Claim 5, further comprising:

determining a total transient upgrade constraint that comprises a total number of individual spaces where an upgrade was assigned.

14. The method of Claim 13, wherein the total transient upgrade constraint is

$$\sum_{i} \sum_{ART>RT(i).} y.AD(i).LS(i).RC(i).RT(i).ART$$

15. The method of Claim 1, further comprising:

determining a space protection constraint that ensures that more individual space and function space than available is not assigned.

16. The method of Claim 15, wherein the space protection constraint is

$$\sum_{i|AD(i)\le SD(i),AD(i)+LS(i)\ge SD,\ ART=RT} y.AD(i).LS(i).RC(i).RT(i).ART + \sum_{OID} \sum_{rt|ART(rt)=RT.} yOpp.OID.SD.RT.ART(RT)$$

17. The method of Claim 5, further comprising:

determining a group space opportunity constraint that ensures that a group opportunity is fully satisfied before being accepted.

18. The method of Claim 17, wherein the group space opportunity constraint is

$$\sum_{ART\ge RT.} yOpp.OID.SD.RT.ART \ge N(OID, SD, RT)wOpp.OID$$

19. The method of Claim 1, further comprising:

determining a group yield constraint that incorporates individual space cost information and function space cost information for a group opportunity.

20. The method of Claim 19, wherein the group yield constraint is

$$\sum_{OID} Y(OID)wOpp.OID$$

wherein Y(OID) is the yield associated with the group demand OID.

21. The method of Claim 1, further comprising:

determining a function space constraint that ensures that a particular function space is not used more than once during a given time period.

22. The method of Claim 21, wherein the function space constraint is

$$\sum_{SS\in C(SS)} sscfu.DP.SS \le 1.$$

23. The method of Claim 1, further comprising:

determining an assigned function space constraint that ensures that an assigned function space is at least as big as a requested function space.

24. The method of Claim 23, wherein the assigned function space constraint is

$$tsa.DP.TST = \sum_{SS \in C(TST).} sscfu.DP.SS$$

25. The method of Claim 5, further comprising:

determining an upgrade function space constraint that ensures that transient upgrades and group upgrades are not given when not necessary.

26. The method of Claim 25, wherein the upgrade function space constraint is

$$\sum_{OID} N(OID, DP, TST)wOpp.OID + \sum_{RTST|ATST=TST} upg.DP.RTST.ATST - \sum_{ATST|RTST=TST} upg.DP.RTST.ATST \leq tsa.DP.TST.$$

27. The method of Claim 5, wherein the bounds comprise at least one step selected from the group consisting of:

setting the amount of accepted transient individual spaces to greater than or equal to 0;

setting the amount of assigned transient individual spaces to greater than or equal to 0;

setting the amount of assigned group individual spaces and group function spaces to greater than or equal to 0;

setting the value of group opportunities to greater than or equal to 0, and less than or equal to 1;

setting the value of group function space greater than or equal to 0, and less than or equal to 1; and

setting the value of space protection to greater than or equal to 0, and less than or equal to the amount of individual space available.

28. A computer system for calculating a potential optimum yield for an entity for a set of demands including both transient demands and group demands comprising:

a processor;

a memory for storing a set of demands, the memory connected to the processor;

wherein the processor is configured to perform the steps of

setting a yield formula for calculating a yield that includes both transient yield from transient individual space and group yield from group individual space and group function space;

determining constraints related to the yield formula;

determining bounds related to the yield formula;

identifying which demands should be accepted in order to maximize the yield, subject to the constraints and the bounds; and

determining the potential optimum yield utilizing the demands identified in the identifying step.

29. A computer program product comprising a computer usable medium having control logic stored therein for causing a computer to provide a user with an application for calculating a potential optimum yield for an entity for a set of demands including both transient demands and group demands comprising:

first computable program code means for setting a yield formula for calculating a yield that includes both transient yield from transient individual space and group yield from group individual space and group function space;

second computable program code means for determining constraints related to the yield formula;

third computable program code means for determining bounds related to the yield formula; and

fourth computable program code means for identifying which demands should be accepted in order to maximize the yield, subject to the constraints and the bounds; and

fifth computable program code means for determining the potential optimum yield utilizing the demands identified in the identifying step.